```
% This code simulated the energy detection in the cognitive radio when the
% sensing channel is Rayleigh flat-fading channel.

%The assumptions are following:
% 1) The primary signal is deterministic and binal phase-shift-keying
% (BPSK).
% 2) Noise is real Gaussian with mean 0 and variance 1.

% The probability of detection for Rayleigh channel can be calculated
% by the averaging the probability of detection for AWGN channel, which is
% given by (1) below.
% Pd_theory_awgn = marcumq(sqrt(L*snr),sqrt(thresh),L./2); % --------(1) The probability
of detection when the channel is AWGN.
% The probability of detection for Rayleigh channel is given by (2), at the
% end of the code.

% Please refer the tutorial paper of the energy detection theory titled
% "Unveiling the Hidden Assumptions of Energy Detector Based Spectrum Sensing for
% Cognitive Radios."

% This code is written by Sanket Kalamkar, Indian Institute of Technology
% Kanpur, India.
% http://home.iitk.ac.in/~kalamkar/

clc
close all
clear all
L = 10; % Number of sensing samples
iter =10^5; % Number of iterations for Monte Carlo Simulation
Pf =0.01:0.03:1; % Probability of False Alarm
snr_db = 0; % Signal-to-noise ratio (SNR) in dB
snr = 10.^(snr_db./10); % SNR in linear scale
for tt = 1:length(Pf) % Calculating threshold for each value of Pf
 energy_fin = []; % Initialization
 n = [];
 y = [];
 energy = [];
 energy_fin = [];

 n=randn(iter,L); % Gaussian noise, mean 0, variance 1
 y = n; % Received signal at the secondary user under the hypothesis H0
 energy = abs(y).^2; % Energy of received signal over L sensing samples under the
hypothesis H0
for kk=1:iter % Start simulation loop to calculate the threhsold
 energy_fin(kk,:) =sum(energy(kk,:)); % Test Statistic of the energy detection
end
energy_fin = energy_fin'; % Taking transpose to arrage values in descending order
energy_desc = sort(energy_fin,'descend')'; % Arrange values in descending order
thresh_sim(tt) = energy_desc(ceil(Pf(tt)*iter));% Threshold obtained by simulations; the
first 'Pf' fraction of values lie above the threshold
tt
```

```
end

%% Simulated probability of detection for Rayleigh channel

for tt = 1:length(Pf)

    s = [];  % Initializtion
    h = [];  % Initializtion


    mes = randi([0 1],iter,L); % Generating 0 and 1 with equal probability for BPSK
    s = (2.*(mes)-1); % BPSK modulation
    h1 = (randn(iter,1)+j*randn(iter,1))./(sqrt(2));% Generating Rayleigh channel
coefficient
    h = repmat(h1,1,L); % Slow-fading is considered, i.e., the channel remains the same
during the senisng process.
    y = sqrt(snr).*abs(h).*s + n; % Received signal y at the secondary user, abs(h) is
the Rayleigh channel gain.
    energy = (abs(y).^2);  % Received energy under the hypotheis H1

for kk=1:iter % Number of Monte Carlo Simulations
    energy_fin(kk) =sum(energy(kk,:)); % Test Statistic for the energy detection under
the hypothesis H1
end

    upper = (energy_fin >= thresh_sim(tt)); % Checking whether the received energy above
the threshold
    i = sum(upper); % Count how many times out of 'iter', the received energy is above
the threshold.
    Pd_sim(tt) = i/kk; % Calculation of the probability of detection (simulated)
tt
end
plot(Pf,Pd_sim,'k')
hold on
%% Probability of detection (Theory): Rayleigh Fading
thresh_theory = 2*gammaincinv(1-Pf,L/2);% Theory threhsold, from the formula of Pf =
gammainc(thresh_theory./(2), L/2,'upper')
temp1 = snr*L/2;
Pd_theory  = [];
A = temp1./(1 + temp1);
u = L./2; % Time-Bandwidth product
for pp = 1:length(Pf)
    n = 0:1:u-2;
    term_sum1 = sum((1./factorial(n)).*(thresh_theory(pp)./2).^(n));
    term_sum2 = sum((1./factorial(n)).*(((thresh_theory(pp)./2).*(A)).^(n)));
    Pd_theory(pp) = exp(-thresh_theory(pp)./2).*term_sum1 + (1./A).^(u-1).*(exp(-
thresh_theory(pp)./(2.*(1+temp1))) - exp(-thresh_theory(pp)./2).*term_sum2);%
---------(2) Theory Probability of detection for Rayleigh
end
 plot(Pf,Pd_theory,'bo') % ROC curve
 hold on
```

```
legend('Simulated P_d','Theory P_d')
xlabel('Probability of False Alarm')
ylabel('Probability of Detection')
```